

Escola Universitària Politécnica de Mataró

Centre adscrit a:



**UNIVERSITAT POLITÈCNICA
DE CATALUNYA**

Enginyeria Tècnica d'Informàtica de Gestió

Aprenentatge de la llengua de signes catalana amb Kinect

Memòria

**Marc Górriz
PONENT: Alfons Palacios**

PRIMAVERA 2012

Agraïments

A Matias Lizana membre del CCI-TCM

A Alfons Palacios tutor del treball

A Anna Rovira intèrpret de llengua de signes catalana

A tots els participants en la captura de mostres

Resum

S'ha desenvolupat una aplicació per a facilitar l'aprenentatge de l'abecedari de la Llengua de signes catalana mitjançant una sèrie d'activitats de lletrejar paraules. El projecte fa ús del dispositiu Kinect per tal de captar les imatges que seran tractades amb les llibreries de codi lliure OpenCV i un algoritme de reconeixement *K nearest neighbors* per tal de portar a terme el reconeixement dels signes.

L'aplicació permet reconèixer un total de sis signes de diferents mides, orientacions i formes.

Resumen

Se ha desarrollado una aplicación para facilitar el aprendizaje del abecedario de la Lengua de signos catalana mediante una serie de actividades de deletrear palabras. El proyecto hace uso del dispositivo Kinect para captar las imágenes que serán tratadas con las librerías de código libre OpenCV y un algoritmo de reconocimiento *K nearest neighbors* para llevar a cabo el reconocimiento de los signos.

La aplicación permite reconocer un total de seis signos de diferentes medidas, orientaciones y formas.

Abstract

Has been developed an Application for learn the Catalan Sign Language alphabet with some spelling activities. The project uses Kinect device to capture images that will be treated with the open source OpenCV library and a recognition algorithm *K nearest neighbors* to carry out the recognition of the signs.

The application can recognize a total of six signs of different sizes, orientations and shapes.

Índex.

Índex de figures.....	III
Índex de taules.	V
Glossari de termes.	VII
1.Introducció.....	1
2. Objectius.	3
2.1. Propòsit.	3
2.2. Finalitat.	3
2.3. Objecte.	3
2.4. Abast.	3
3. Estudi Previ.	5
3.1. Llengua de signes.....	5
3.2 Metodologia Empírica.....	9
3.2.1. Signes 9	
3.2.2.Captura de les imatges..... 10	
3.2.3 Binarització..... 11	
3.2.4. Mostres. 11	
3.2.5. Marcadors. 13	
3.2.6. Algoritmes de reconeixement..... 15	
3.3. OpenCV 17	
3.3.1 EmguCV 17	
3.3.2 Usos 17	
3.4. Kinect. 18	
3.4.1. Components 18	
3.4.2. Funcions 19	
3.4.3. Percepció de la profunditat. 21	
3.4.4 Alternatives..... 21	
3.4.5 Avantatges 21	
3.4.5. Kit de desenvolupament. 22	

4. Anàlisi	25
4.1. Planificació	25
4.2. Requeriments	25
4.3. Casos d'ús	26
4.4. Diagrama d'activitat	29
4.5. Diagrama de classes	30
4.6. Metodologia de treball	31
5. Disseny	33
5.1. Disseny de la persistència	33
5.2. Disseny de la interfície	34
6. Desenvolupament	39
6.1. Configuració de l'entorn de treball	39
6.1.1. Instal·lació dels drivers	39
6.1.2. Instal·lació de les llibreries	39
7. Resultats	41
7.1. Ratis d'encert	41
7.2. Mètodes de correcció	42
8. Problemes trobats	43
9. Conclusions	45
10. Futures Ampliacions	47
11. Referències:	49

Índex de figures.

Fig. 3.1 Alfabet dactilològic [2]	8
Fig. 3.2 Metodologia empírica	9
Fig. 3.3 Dispositiu Kinect	18
Fig. 3.4 Detecció de moviment [5]	19
Fig. 3.5 Seguiment esquemàtic [5]	20
Fig. 3.6 Reconeixement facial [5]	20
Fig. 3.7 Detecció de la profunditat	21
Fig. 4.1 Diagrama de casos d'ús	26
Fig. 5.1 Tag caràcter del xml de preselecció	34
Fig. 5.2 Menú principal	34
Fig. 5.3 Tutorial calibració	35
Fig. 5.4 Opcions	36
Fig. 5.5 Pantalla Calibració	36
Fig. 5.6 Estímul	37
Fig. 5.7 Menú Avançat	38

Índex de taules.

Taula 3.1. Comparativa SDK.....	23
---------------------------------	----

Glossari de termes.

ASL	Llengua de Signes Americana
Binaritzar	Sistema que modifica la coloració de la imatge classificant tots els píxels de la imatge en dos valors, blanc o negre, obtenint així una imatge en blanc i negre.
Body-tracking	Sistema que detecta i segueix diferents punts del cos en l'espai, permetent la interpretació de postures i moviments.
Estímul	Mots o imatges a lletrejar mostrades durant l'execució de l'aplicació
FPS	Frames per segon, nombre d'imatges que es mostren/capturen en un segon.
KNN	K veïns més propers (<i>k nearest neighbors</i>), sistema de reconeixement basat en la premissa que els elements de la mateixa classe tindran valors de marcador més propers.
LSC	Llengua de Signes Catalana
LSE	Llengua de Signes Espanyola
Píxel	Menor unitat que forma una imatge digital. Conté informació sobre la tonalitat del color.
RAW	Format d'imatge que conté la totalitat de les dades de la imatge.
RGB	Descomposició d'una imatge en tres colors; vermell, verd i blau (<i>red, green, blue</i>).
YCrCb	Descomposició d'una imatge en component de luminància (Y) que determina si el color es clar o fosc, component blau (Cb) que determina la quantitat de color blau i component vermell (Cr) que determina la quantitat de color vermell.

VS Visual Studio entorn de desenvolupament de software desenvolupat per Microsoft.

Wrapper Funció que s'encarrega de cridar la funció que realment duu a terme el procés. Fent una funció de interfície o pont entre la funció real y l'usuari.

1.Introducció

En una societat on cada dia s'aposta més per la integració de les persones amb discapacitat i on la tecnologia cada dia és més present, cal cercar formes d'aprofitar-la per aquest fi.

Aquest projecte construeix una eina per facilitar l'aprenentatge i la millora en l'ús de la llengua de signes catalana (LSC).

L'any 2010, el Parlament de Catalunya va aprovar per llei les regulacions, l'estudi, la promoció i els requeriments dels intèrprets de la LSC.

La tecnologia escollida és el dispositiu Kinect desenvolupat per Microsoft mitjançant el qual es capturen els signes duts a terme per l'usuari per tal de poder realitzar una sèrie d'exercicis.

El procés de captura es duu a terme utilitzant la càmera de profunditat que inclou Kinect. Quan s'ha detectat la mà s'emmagatzema una imatge binaritzada.

La imatge binaritzada ha de passar un sistema de preselecció que s'encarrega de descartar les lletres que no coincideixin en orientació i nombre de vèrtexs amb la imatge d'entrada.

Després de la preselecció s'inicia l'algoritme de Knn, aquest algoritme utilitza dos paràmetres que són l'àrea de la mà segmentada en quatre porcions i el perímetre de la mà.

L'algoritme Knn determina la lletra que ha realitzat l'usuari comparant el valor dels dos marcadors de la mostra realitzada per l'usuari amb les mostres d'entrenament que té l'aplicació.

Les mostres d'entrenament són imatges dels signes realitzades per un grup de persones de diferents edats, sexes i races.

2. Objectius.

2.1. Propòsit.

Estudiar i realitzar totes les etapes necessàries per al reconeixement de les lletres de la LSC. A més de desenvolupar l'aplicació per aprendre la LSC mitjançant uns exercicis, que consisteixen en lletrejar paraules mostrades a l'usuari de forma escrita o mitjançant una imatge.

2.2. Finalitat.

Facilitar l'aprenentatge de la llengua de signes catalana gràcies a una sèrie d'exercicis bàsics basats en lletrejar paraules. Proporcionar una base per al desenvolupament d'eines de caràcter similar.

2.3. Objecte.

Aplicació que reconeix els gestos que conformen l'abecedari català de signes mitjançant el dispositiu Kinect, permetent realitzar una sèrie d'activitats per lletrejar la LSC a partir del que es presenta en un text o imatge.

2.4. Abast.

Tot el procés de reconeixement des de què es captura la imatge, passant per la localització i el calibratge de la mà, la obtenció dels diferents paràmetres del signe i per últim realitzant una comparació amb els gestos presents a les mostres de l'aplicació, per tal de reconèixer el signe concret.

3. Estudi previ.

3.1. Llengua de signes.

La llengua de signes és el resultat de la mútua interacció entre biologia i cultura en l'ésser humà; és una adaptació creativa a una limitació sensorial, permetent aprofitar la gran expressivitat del cos humà per a la comunicació a través d'una modalitat visual.

3.1.1. Història

L'estudi de les llengües de signes es remunta a la dècada dels seixanta, concretament el 1960, en què William C. Stockoe, un lingüista nord-americà, publica *Sign Language Structure* (Estructura de la llengua de signes), el primer treball que aprofundeix en l'anàlisi de les llengües de signes fent servir els coneixements de la lingüística moderna.

L'aparició d'aquesta obra suposa una revolució del concepte que es tenia sobre les llengües de signes fins a aquell moment, que no passava d'una senzilla descripció de la llengua com un conjunt de gestos pactats a partir de les regles de la llengua oral. Posteriorment a aquesta publicació, que Stockoe complementà cinc anys més tard amb la publicació de *A Dictionary of American Sign Language on Linguistic Principles* (Un diccionari de la llengua de signes americana en els principis de la lingüística).

L'Úrsula Bellugi i l'Edward S. Klima presenten la seva obra principal *The Signs of Language* (Els signes de la llengua) que aprofundeix en els aspectes arbitraris i icònics dels signes i en la seva estructura interna.

D'aquesta manera i gràcies a aquests treballs, les llengües de signes entren a formar part dels interessos de la lingüística. Les primeres passes es van fer als Estats Units amb la llengua de signes americana i progressivament aquest interès es va anar estenent arreu del món a partir dels anys 80.[1]

3.1.2. Estructura

L'estructura de les llengües de signes té els mateixos nivells que les llengües orals, malgrat les diferències que existeixen pel que fa al canal de transmissió. La seva eficàcia com a instrument per al desenvolupament lingüístic està demostrada, la qual cosa comporta alhora un desenvolupament cognitiu total de la persona sorda que li possibilita viure les seves relacions socials i emocionals de manera plena. Els estudis duts a terme aquests darrers 50 anys permeten analitzar la dimensió fonològica, morfològica, sintàctica i discursiva de les llengües de signes de la mateixa manera que es pot fer amb qualsevol llengua oral. [1]

3.1.3. Situació de la llengua de signes catalana.

El domini lingüístic de la LSC no es correspon amb el del català oral, ja que els territoris d'aquest domini són Catalunya, la Franja d'Aragó i Menorca, això sí, amb un percentatge de signants de gairebé el 100%. En canvi, l'LSE es fa servir a la resta de territoris de l'Estat amb varietats geogràfiques diferents segons el territori. Així, tot i que el País Valencià, Euskadi i Galícia tenen llengües orals pròpies cooficials amb el castellà, la llengua de signes que s'hi utilitza és l'LSE. Senzillament, no hem d'establir una relació directa entre la llengua oral d'un territori i la llengua de signes perquè són llengües de naturalesa i història ben diferents.

La LSC és patrimoni cultural i lingüístic de Catalunya i dels altres territoris on s'utilitza i ha de ser respectada i fomentada com qualsevol altra llengua natural del territori. La configuració de la LSC com a llengua pròpia diferenciada es remunta a uns 200 anys de trajectòria, al llarg dels quals es va anar desenvolupant paral·lelament a la LSE tot conservant cadascuna els seus trets lingüístics propis. Les diferències lingüístiques de la LSC tenen el seu origen en les escolaritzacions dels nens sords en règim d'internament, que van propiciar, al voltant de dos segles enrere, que la llengua evolucionés en una direcció determinada sota la influència d'un entorn cultural aïllat. Actualment a Catalunya,

existeixen uns 25.000 usuaris de l'LSC, dels quals 6.000 són persones sordes i sordcegues i la resta, uns 19.000, són oïdors.

Les persones sordes, a través del moviment associatiu sord, han lluitat al llarg de la història perquè es reconegui la llengua de signes i se'n normalitzi l'aprenentatge, la docència, l'ús i la recerca que es mereix. No va ser fins al 19 de juny de 2007 que la Comissió de Treball i Afers Socials del Congrés dels Diputats va aprovar el projecte de llei pel qual es donava suport explícit a totes les llengües de signes de l'Estat espanyol, ratificat a l'octubre pel Senat, de manera que les llengües de signes de l'Estat quedaven reconegudes i es regulaven els mitjans de suport a la comunicació de les persones sordes, amb discapacitat auditiva i sordcegues (Llei 27/2007 de 23 d'octubre. BOE núm. 255,18476).

Darrerament, el 26 de maig de 2010, el Parlament va aprovar per unanimitat la Llei de la llengua de signes catalana, que rep actualment "la dignitat que li pertoca a través del seu reconeixement institucional". La nova norma dóna resposta al mandat que estableix l'Estatut en el seu article 50.6 que determina que s'ha de garantir l'ús de la llengua de signes catalana, el seu ensenyament i protecció i vetllar per la igualtat de les persones amb sordesa que l'usin. En definitiva, la norma regula el dret d'ús, l'aprenentatge, la docència, la recerca i la interpretació. En relació a la recerca, es responsabilita de l'Institut d'Estudis Catalans (IEC) i de les universitats del país. També compromet l'administració a difondre i fomentar l'aprenentatge de la llengua entre la ciutadania i entre el personal de les administracions. El text reconeix l'IEC com l'autoritat normativa i preveu també la creació del Consell Social de la llengua de signes catalana, com a òrgan d'assessorament, consulta i participació. [1]

3.1.4. Alfabet dactilològic




	A		B		C		Ç
	D		E		F		G
	H		I		J		K
	L		M		N		O
	P		Q		R		S
	T		U		V		W
	X		Y		Z	alfabet dactilològic	

Fig. 3.1 Alfabet dactilològic [2]

Cadascuna de les lletres de l'alfabet de la llengua catalana té un signe que la representa en la LSC. El conjunt d'aquests signes es denomina "alfabet dactilològic".

El "alfabet dactilològic" català està format per 27 signes dels quals 18 són estàtics i 9 són dinàmics (amb moviment), la única diferència d'aquest amb el castellà es que conté la Ç enlloc de la Ñ.

3.2 Metodologia Empírica

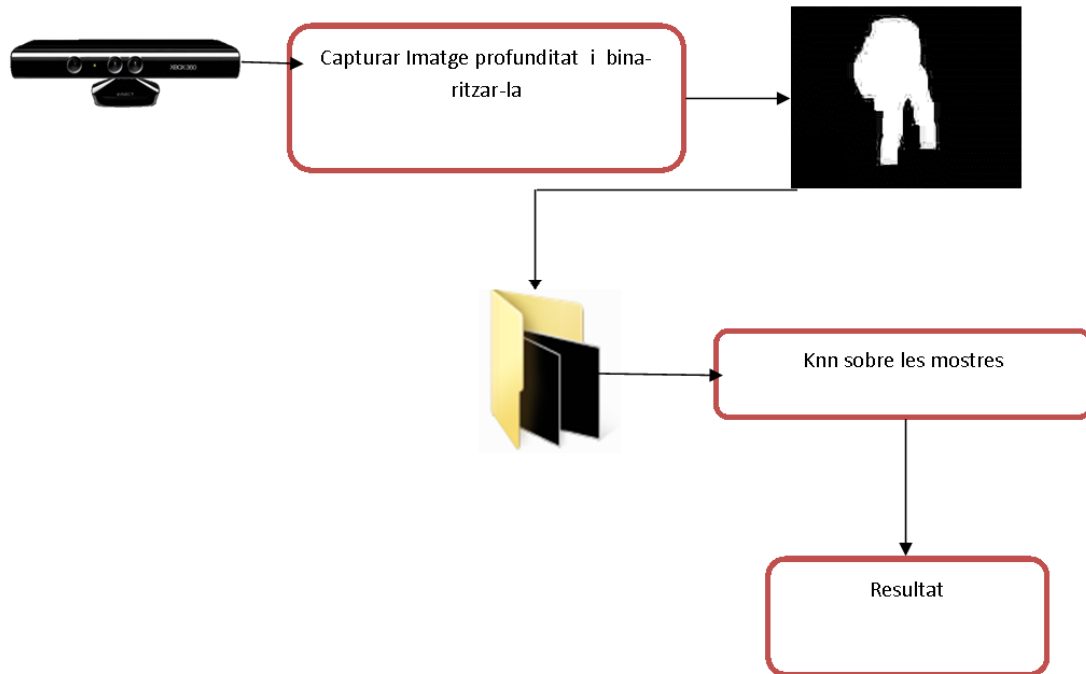


Fig. 3.2 Metodologia empírica

3.2.1. Signes

En aquest projecte només s'ha realitzat el reconeixement de sis signes degut a que els signes dinàmics s'han deixat fora del abast d'aquest projecte i afegir més lletres estàtiques no aporta cap característica nova al projecte.

Inicialment es van escollir tres lletres A, B i N. A i B per a poder provar l'eficàcia del marcador àrea per tal de distingir dos lletres similars però amb diferent mida. N es va escollir per a provar la capacitat del perímetre per diferenciar entre B i N lletres amb un àrea similar però perímetre diferent.

En una fase més avançada del projecte es van afegir tres lletres més per tal d'avaluar l'escalabilitat del projecte a més d'altres paràmetres com la capacitat de diferenciar lletres segons l'orientació degut a que la lletra M és idèntica en tots els altres marcadors a B.

Es va afegir la lletra I per tal de provar el nivell de precisió degut a ser molt propera en tots els marcadors a A i per últim, la L per tal de provar una lletra amb una orientació difícil de classificar.

3.2.2. Captura de les imatges

El sistema de captura es realitza a 24 frames per segon i s'utilitza tant per la captura de les mostres d'entrenament, com per la captura durant l'ús del programa.

La captura es pot realitzar mitjançant dos mètodes:

Càmera RGB, aquesta càmera detecta els colors per tant es busca la mà mitjançant una cerca del objecte de major mida del color de la pell.

Càmera de profunditat, càmera que es capaç de detectar la distància a la que es troba cada objecte per tant es defineix una distància màxima i una distància mínima en la que es busca la mà.

La distància es representa mitjançant una imatge amb diferents colors. Per seleccionar el rang s'usa un rang de colors de l'espai YCRCB ja que aquesta escala permet definir millor l'escala de blaus gràcies a que les seues components son la luminància, color vermell i color blau.

Dels dos mètodes, l'aplicació usa el sistema de captura mitjançant la càmera de profunditat perquè aquest sistema només presenta el problema de com evitar que l'usuari introdueixi altres parts del cos en el rang de profunditats. Aquest problema es resol amb una calibració prèvia on l'usuari s'ha de col·locar amb el braç estirat a una distància concreta i fer coincidir la mà amb un contorn.

En canvi amb la càmera RGB si bé no caldria calibració, es requeriria que no hi hagués cap més element de color similar a la pell en tota l'àrea de captura de la càmera.

Aquest fet implica que l'usuari hagi de portar roba de color diferent, de màniga llarga i treure tots els elements de color similar del tir de càmera. Per tant és molt difícil aconseguir les condicions òptimes, el que porta a descartar aquest mètode.

3.2.3 Binarització

Quan ja s'ha detectat la posició on se situa la mà, es procedeix a la binarització que consisteix en convertir la imatge captada pel dispositiu Kinect en una imatge en blanc i negre on es representa la mà en color blanc i tota la resta en color negre.

El procés es realitza recorrent tots el píxels de la imatge original. Si un píxel correspon al color escollit es converteix en un píxel blanc en cas contrari un píxel negre.

Aquest procés es duu a terme perquè sobre una imatge binaritzada es més fàcil processar els diferents paràmetres de la mà.

3.2.4. Mostres.

Les mostres són les imatges de referència que utilitza el sistema per realitzar el reconeixement. Estan compostes per imatges de vuit subjectes diferents entre ells una experta en llengua catalana de signes.

S'usen 70 mostres amb una igual aportació de cada subjecte per signe amb les següents característiques.

Les mostres són imatges JPG en blanc i negre on es representa la mà de color blanc sobre un fons negre amb la resolució nativa de Kinect 640 *480 píxels i 8 bits per píxel al ser una imatge binaritzada.

3.2.4.1. Captura de les mostres

Per a realitzar les captures s'ha escollit una població de vuit persones de diferents edats i sexes per a obtenir una mostra el més variada possible.

Cada persona ha hagut de realitzar cada lletra diverses vegades ja que les persones no sempre realitzen el signe de igual manera.

Aquestes captures es guarden binaritzades per a la seva selecció.

3.2.4.2. Selecció de les mostres

La selecció de les mostres es divideix en dues etapes:

-Etapa 1: Es descarten les mostres amb errors de captura com poden ser elements no desitjats a la imatge o que la mà no es visualitzi correctament. Aquestes mostres defectuoses es descarten ja que si no es pot extreure correctament els marcadors de la mà només aporten soroll a la mostra.

-Etapa 2: Se selecciona només una captura de cada dotze, si la càmera realitza 24 captures per segon s'obtenen dues captures per segon ja que interessa que hi hagi varietat entre les captures. Aquesta etapa es realitza per aconseguir un conjunt de mostres més variat evitant les mostres consecutives.

-Etapa 3: Prenent com a referència les mostres de l'experta i establint una distància màxima es filtren les mostres exteriors a aquest radi per eliminar les mostres que presentin una variació massa elevada respecte els signes de l'expert. Aquesta etapa es realitza per tal d'evitar incloure mostres que augmentin massa la dispersió de la mostra.

Etapa4: Sobre el conjunt filtrat de cada subjecte es duu a terme una tria aleatòria de X mostres obtenint així una mostra equilibrada amb la mateixa quantitat de mostres per subjecte.

3.2.5. Marcadors.

Els marcadors s'han classificat en dos tipus:

Marcadors generals són els marcadors que es basen en característiques de la mà alhora de realitzar el signe però independents de la posició en l'espai com per exemple el perímetre.

Marcadors vectorials que reconeixen els signes mitjançant posicions dels elements de la mà com la posició del canell, les puntes dels dits, el centre del palmell o la separació entre els dits.

En l'aplicació s'han usat únicament marcadors de caràcter general ja que aquests s'adapten correctament a qualsevol signe.

D'altra banda, els marcadors vectorials que són útils per a uns signes a causa de la postura amb la que es realitzen i a ser un reconeixement en dues dimensions són incompatibles amb altres. La dificultat per trobar marcadors vectorials útils per a tots els signes ha provocat que es rebutgi el seu ús.

Els marcadors generals estudiats són els següents:

- Perímetre:

Longitud del contorn de la mà.

Útil per a diferenciar dits junts o separats i signes més grans dels més petits.

- Àrea:

Superfície de la mà en píxels.

Útil per a diferenciar entre signes grans o petits, però descartat degut a que aquesta diferenciació no és suficient per contribuir en el reconeixement, ja que la majoria de signes tenen una mida similar.

- Àrea segmentada:

Es defineix un rectangle al voltant de la mà, dividit en quatre porcions iguals i es calcula l'àrea del signe compresa en cada porció.

Permet una visió més acurada de l'àrea permetent diferenciar entre figures amb un àrea similar si aquesta està distribuïda de forma diferent.

- Centre de gravetat:

És el punt d'aplicació de les diverses forces de gravetat que actuen sobre la distribució de masses d'un cos. La posició del centre de gravetat en les diferents figures es molt similar dificultant obtenir informació útil per al reconeixement.

Valorant els quatre marcadors anteriors s'ha determinat que només dos d'ells són realment útils en el reconeixement. L'àrea segmentada i el perímetre.

S'han descartat l'àrea i el centre gravitatori ja que l'àrea no aporta cap informació valuosa respecte els altres dos marcadors. I el centre gravitatori no aporta informació que permeti diferenciar els signes de forma clara.

Això s'ha determinat realitzant el reconeixement d'un conjunt fix d'imatges per cada lletra amb totes les combinacions de marcadors. Comparant els ratis d'encert queda palès que tant l'àrea com el centre gravitatori només mantenen o redueixen el rati d'encert però no l'augmenten.

3.2.6. Algoritmes de reconeixement

La detecció per Knn consisteix en aconseguir realitzar el reconeixement en base a una sèrie d'imatges d'entrenament de les quals s'extreu una sèrie de valors i en base a un paràmetre K es determina quines mostres són les més semblants.

S'ha seleccionat l'algoritme de Knn per que és un algoritme molt potent. Permet un gran nivell de personalització amb la possibilitat d'afegir marcadors i mostres propis, a més de ser senzill per a ampliacions i millores.

3.2.6.1. Constant K.

El valor K determina el nombre de veïns més propers que s'avaluen per a obtenir el resultat del Knn.

Com més gran sigui K teòricament s'obté un millor resultat ja que s'està basant la resposta en un major nombre de mostres, però això es veu limitat pel nombre de mostres totals. L'augment de K també afecta al rendiment però això cal augmentar-la només si aporta una millora al reconeixement.

S'utilitza una constant $K=10$ degut a que un nombre més gran amb la quantitat de mostres usades provocaria incloure mostres errònies, si fos més petita el resultat es podria veure molt afectat per les variacions en les mostres.

3.2.6.2 Knn

L'algoritme de Knn consta de les següents parts:

Inicialment es calcula el valor dels marcadors de la mostra a reconèixer i de totes les mostres d'entrenament, guardant-los en objectes de la classe Study.

A continuació es calcula la distància euclídea de la mostra a reconèixer amb totes les mostres d'entrenament utilitzant els valors dels marcadors dividits entre la seva variança per tal d'igualar el seu pes; si no es divideix entre la variança els marcadors amb major variança tenen un pes major en la distància.

Obtenir el conjunt de les K distàncies menors i determinar quina lletra té un major nombre d'ocurrències. La lletra predominant és el resultat del reconeixement.

El procés de Knn es repeteix 25 vegades per signe a partir de 50 mostres agafades no consecutivament per reduir l'efecte de possibles errors de captura. D'aquestes 25 es comprova que com a mínim hi hagi una lletra que hagi estat triada mes cops que el valor definit pel marge d'acceptació. El valor òptim del marge d'acceptació s'ha considerat de quinze mitjançant proves amb diferents persones utilitzant l'aplicació. Amb un valor major de quinze no es resol la sensibilitat a errors de captura ja que amb un mínim error no passaria el marge i amb un valor menor de la meitat no es pot garantir que el reconeixement sigui satisfactori.

3.2.6.3. Filtratge de caràcters Knn

Per a millorar l'efectivitat del l'algoritme de reconeixement Knn i optimitzar el temps d'execució abans d'iniciar-se l'algoritme s'executa un filtratge previ.

Aquest filtratge previ es realitza en base a dos paràmetres emmagatzemats en un fitxer XML calculats a partir de les mostres.

- Orientació classifica els signes en tres grups: Horitzontals , verticals o nuls. Aquesta classificació es duu a terme aproximant el contorn de la mà a un rectangle. Si el rectangle és un 25% més alt es considera el signe vertical, en cas de ser un 25% més ample que alt es considera horitzontal i si no es dona cap dels dos casos anteriors serà nul.
- Nombre de vèrtexs Aproximant el contorn de la mà a un polígon i comptant el nombre de vèrtexs d'aquest es pot classificar les figures.

3.3. OpenCV

OpenCv(*Open Source Computer Vision Library*) és una llibreria de funcions de programació enfocada a processament d'imatges en temps real, desenvolupada inicialment per Intel i actualment, amb el suport de Willow Garage, sota la llicència open source BSD license. La llibreria és lliure tant per ús comercial com acadèmic.

La versió 1.0 va ser llançada l'any 2006 i l'última versió és la 2.4 que data de l'1 de maig del 2012.

OpenCv està disponible per a C++, C i Python, però existeixen wrappers que permeten utilitzar les llibreries des d'altres llenguatges.[3]

3.3.1 EmguCV

EmguCv són un conjunt de llibreries de codi obert que permeten l'ús d'OpenCv des de tots els llenguatges compatibles amb .NET com C# o Visual Studio entre d'altres.

La funció de la llibreria és la d'un wrapper, es basa en fer de pont entre aquests llenguatges i l'OpenCv creant un mètode i una estructura C# per a cadascun dels mètodes i estructures originals.[4]

3.3.2 Usos

Open CV presenta una gran varietat de mètodes pensats per a projectes dins dels següents camps:

Eines per a tractament 2D i 3D

Sistemes de reconeixement facial

Reconeixement de gestos

Interacció persona ordinador

Robòtica

Comprensió de moviments

Identificació d'objectes

Visió Estèreo: Captura d'imatges amb dues càmeres

Captura de moviment

3.4. Kinect.

Kinect és un sensor de moviment creat per Microsoft per a la videoconsola Xbox 360 i PC. Kinect permet als usuaris controlar i interactuar amb la videoconsola sense necessitat d'usar un controlador mitjançant control per veu i gestos corporals.

El dispositiu es va llançar al mercat a Estats Units el 4 de novembre del 2010 i a Europa el dia 10 del mateix mes.

3.4.1. Components

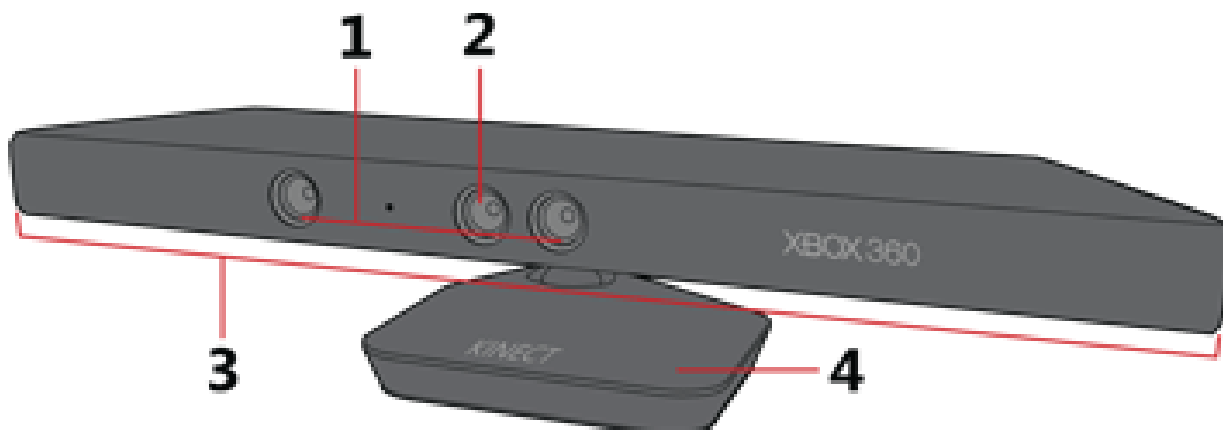


Fig. 3.3 Dispositiu Kinect

1. Sensors de profunditat 3-D

Els sensors tridimensionals fan un seguiment de cos dins l'àrea de joc.

2. Càmera RGB

Una càmera RGB (vermell, verd, blau) ajuda a la identificació, ja que permet captar els colors i capta imatges i vídeos del joc.

3. Diversos micròfons

S'usen un conjunt de micròfons situats al frontal inferior del sensor Kinect utilitzats per a la captació de sons i el reconeixement de veu.

4. Inclinació motoritzada

Un mecanisme a la base del sensor Kinect inclina de manera automàtica el sensor cap amunt o cap a baix segons sigui necessari.

3.4.2. Funcions

Sensor de moviment

Detecta la posició en l'espai de diferents punts del cos d'una o dues persones simultàniament.



Fig. 3.4 Detecció de moviment [5]

Seguiment esquemàtic

Basant-se en les dades de profunditat és capaç de detectar desplaçaments en l'espai o salts.



Fig. 3.5 Seguiment esquemàtic [5]

Reconeixement facial

A través de la captura d'una sèrie de trets facials és capaç de diferenciar entre diferents persones.



Fig. 3.6 Reconeixement facial [5]

3.4.3. Percepció de la profunditat.

El sensor actua com un sònar a partir del temps de sortida i arribada de la llum després de reflectir-se en un objecte; sabent la velocitat absoluta de la llum, es pot obtenir la distància a la qual es troba aquest objecte. En un ampli camp visual amb objectes, la càmera Kinect tracta de reconèixer a quina distància estan els objectes, distingint el moviment en temps real. Kinect pot arribar a distingir la profunditat de cada objecte amb diferències de un centímetre i la seva altura i amplària amb diferències de tres mil·límetres.

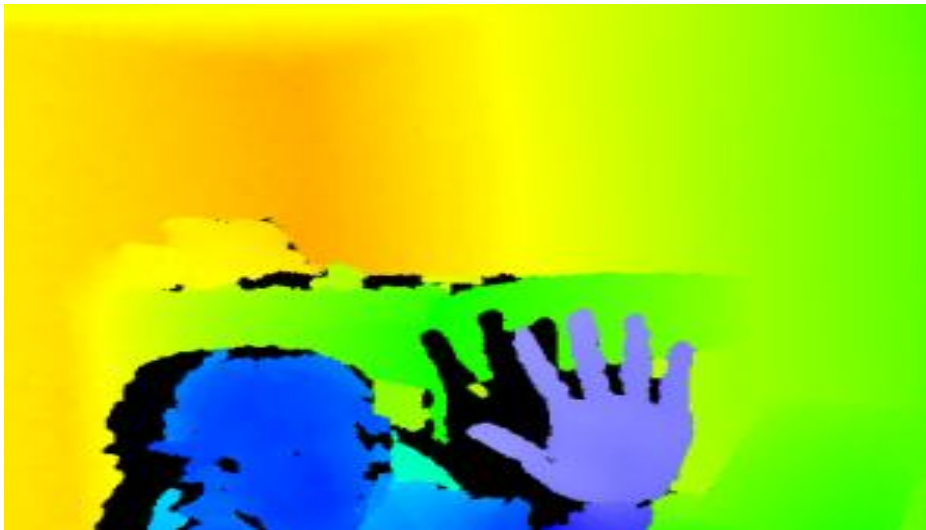


Fig. 3.7 Detecció de la profunditat

3.4.4 Alternatives

Existeixen alternatives a la càmera de Microsoft com són les Webcams i l' ASUS Wavi Xtion un dispositiu de característiques idèntiques a les del Kinect, ja que ambdós utilitzen la tecnologia desenvolupada per la empresa PrimeSense [6].

3.4.5 Avantatges

Tal com s'ha presentat en els punts anteriors el dispositiu desenvolupat per Microsoft presenta funcionalitats afegides a les que proporciona una càmera web estàndard.

En el desenvolupament del reconeixement de signes de la LSC no es pot aprofitar la detecció pròpia de Kinect ja que només esta pensada per detectar posicions de cos sencer però no els gestos de la pròpia mà.

Els avantatges que es tenen són els derivats d'utilitzar una càmera d'infrarojos enlloc d'una càmera RGB tradicional. Aquests avantatges són:

- **Major independència de l'entorn:** No afecten els colors ni la il·luminació del lloc on es desenvolupa el reconeixement.
- **Major independència del subjecte:** No afecta el color de la pell ni la roba del subjecte que realitza la prova.
- **Control de la profunditat:** Permet detectar la profunditat a la que es troba cada objecte podent així limitar un rang de profunditats acceptable, per tal d'evitar errors amb la resta del cos o amb el fons de la imatge.

Respecte al dispositiu Wavi Xtion d'Asus no existeix cap diferència remarcable ja que és un dispositiu igualment vàlid al Kinect per a dur a terme aquest projecte.

3.4.5. Kit de desenvolupament.

En el desenvolupament de Rehabilitació amb Kinect són necessaris:

- Drivers i llibreries per a la Kinect.
- Entorn / llenguatge de desenvolupament.

3.4.5.1. Drivers i llibreries per a la Kinect.

Els drivers són un conjunt d'arxius amb la configuració necessària per a què el sistema operatiu pugui reconèixer un dispositiu. Les llibreries són un conjunt d'arxius capaços de

proporcionar ús al dispositiu. Els més comuns per Kinect són:

- Microsoft SDK.

El SDK oficial desenvolupat per Microsoft inclou llibreries de body-tracking per detectar els diferents punts del cos en l'espai i llibreries per al reconeixement de veu [7].

- CL NUI Platform.

CL NUI Platform és una solució auto instal·lable i de codi obert per Kinect de molt fàcil instal·lació i utilització. Disponible únicament per Windows i amb llibreries precompilades. Code Laboratories és l'empresa especialitzada en programari de codi obert encarregada de desenvolupar aquest driver i d'altres [8].

	CLNUI	Microsoft SDK
Llicència	Gratuïta	Gratuïta només per ús no comercial
Documentació	Poca	Molta
Detecció de dits	No	No
SO	Windows XP/Vista/7	Només Windows 7
Facilitat d'instal·lació	Sí	Sí
Detecció de l'esquelet	No	Sí
Projectes desenvolupats amb ell	Bastants	Pocs, degut a ser més recent

Taula 3.1. Comparativa SDK.

3.4.5.2. Llenguatges de desenvolupament

- **C#** és un llenguatge de programació orientat a objectes desenvolupat i estandarditzat per Microsoft com a part de la seva plataforma .NET, que després va ser aprovat com un estàndard per la ECMA (ECMA-334) i ISO (ISO/IEC 23270). C# és un dels llenguatges de programació dissenyats per a la infraestructura de llenguatge comú. La seva sintaxi bàsica deriva de C/C++ i utilitza el model d'objectes de la plataforma .NET, similar al de Java, encara que inclou millores derivades d'altres llenguatges.

- **C++** és un llenguatge de programació que fou creat, com el seu predecessor C, als laboratoris Bell d' AT&T. El seu autor principal va ser Bjarne Stroustrup. L'any 1980 es van afegir noves característiques al llenguatge C, entre les principals la integració de les classes, idea que va ser presa de Simula67 (per molts considerat el primer llenguatge orientat a l'objecte). A partir d'aquí va anar evolucionant fins que l'any 1985 va ser consolidat com un llenguatge orientat a objectes i anomenat C++.

4. Anàlisi

4.1. Planificació

La planificació temporal està establerta amb la següent distribució d'hores:

- Recopilació d'informació i estudi previ: 60 hores
- Desenvolupament: 200 hores
- Proves i correccions: 80 hores
- Redacció de la documentació: 50 hores

4.2 Requeriments

Aplicació per a ordinadors amb SO Windows i el dispositiu Kinect que permeti lletrejar mitjançant la LSC el contingut de les imatges que es mostren per pantalla. L'aplicació ha de servir tant per aprendre les lletres com per practicar-les.

El conjunt de lletres a reconèixer és limitat a les lletres estàtiques que no presenten cap tipus de moviment. No es considera prioritària la inclusió de la totalitat de les lletres estàtiques però es necessari que l'algoritme de reconeixement sigui robust per a poder-les suportar en un futur.

Cal garantir que el sistema de reconeixement funciona amb diferents persones.

El sistema ha d'estar preparat per usuaris avançats que vulguin realitzar recerca, per tant s'ha de poder modificar els paràmetres que utilitza l'aplicació com el valor de K , les mostres d'entrenament i el marge d'acceptació.

És prioritari minimitzar el temps necessari per realitzar el reconeixement ja que s'usa en una aplicació de temps real.

4.3 Casos d'ús

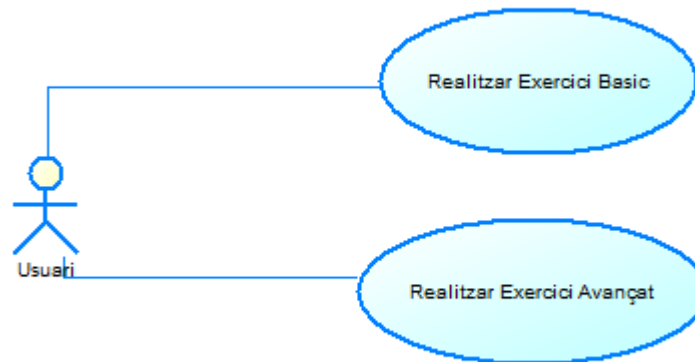


Fig. 4.1 Diagrama de casos d'ús

Realitzar exercici (bàsic)

Cas d'ús que consisteix en realitzar la activitat de lletrejar i el seu procés previ, amb els paràmetres (K, número de mostres per lletra, marge d'acceptació i marcadors) per defecte.

L'usuari inicia el cas d'ús. El sistema mostra la pantalla de tutorial de la calibració on s'explica la postura i la distància correctes.

L'usuari comença la calibració i el sistema mostra la pantalla de calibració que l'usuari realitza fent coincidir la seva mà amb un contorn mostrat per pantalla. En aquesta pantalla l'usuari pot seleccionar opcions per modificar la inclinació del dispositiu Kinect i el temps que deixa el sistema a l'usuari per a realitzar el signe.

A continuació el sistema mostra l'estímul de la tasca senyalant la lletra a signar, en aquest moment l'usuari pot realitzar diferents accions:

- L'usuari accedeix a tutorial, el sistema el mostra i l'usuari finalitza el tutorial retornant a la situació anterior.

- L'usuari accedeix a ajuda, el sistema el mostra i l'usuari finalitza l'ajuda retornant a la situació anterior.
- L'usuari realitza un signe, el sistema el processa i el sistema proporciona el feedback provocant diferents esdeveniments:
 - La lletra és correcta i ja s'han realitzat totes les lletres i tots els estímuls, finalitza el cas d'us.
 - La lletra és correcta i ja s'han realitzat totes les lletres, el sistema mostra el següent estímulo.
 - La lletra és correcta i queden lletres en el mateix estímulo, el sistema passa a la següent lletra.
 - La lletra és incorrecta, el sistema retorna al estímulo assenyalant la mateixa lletra.

Realitzar l'exercici (avançat)

Cas d'ús que consisteix en realitzar la activitat de lletrejar i el seu procés previ amb els paràmetres (K, numero de mostres per lletra , marge d'acceptació i marcadors) seleccionats per l'usuari.

Precondició: El nombre de mostres seleccionat ha de ser menor o igual al nombre de mostres emmagatzemat per cada lletra a la persistència.

- 1.L'usuari inicia els cas d'us.
- 2.El sistema demana el valor del paràmetre K, el nombre de mostres i els marcadors a utilitzar.
- 3.L'usuari introdueix la informació sol·licitada.
- 4.S'inicia el cas d'ús realitzar exercici bàsic amb els paràmetres entrats per l'usuari, si algun és erroni s'executa amb el valor per defecte.

4.4 Diagrama d'activitat

Diagrama d'activitat que mostra el procés de reconeixement dut a terme pel sistema quan l'usuari realitza el signe ens els casos d'us realitzar exercici (bàsic) i realitzar exercici (avançat).

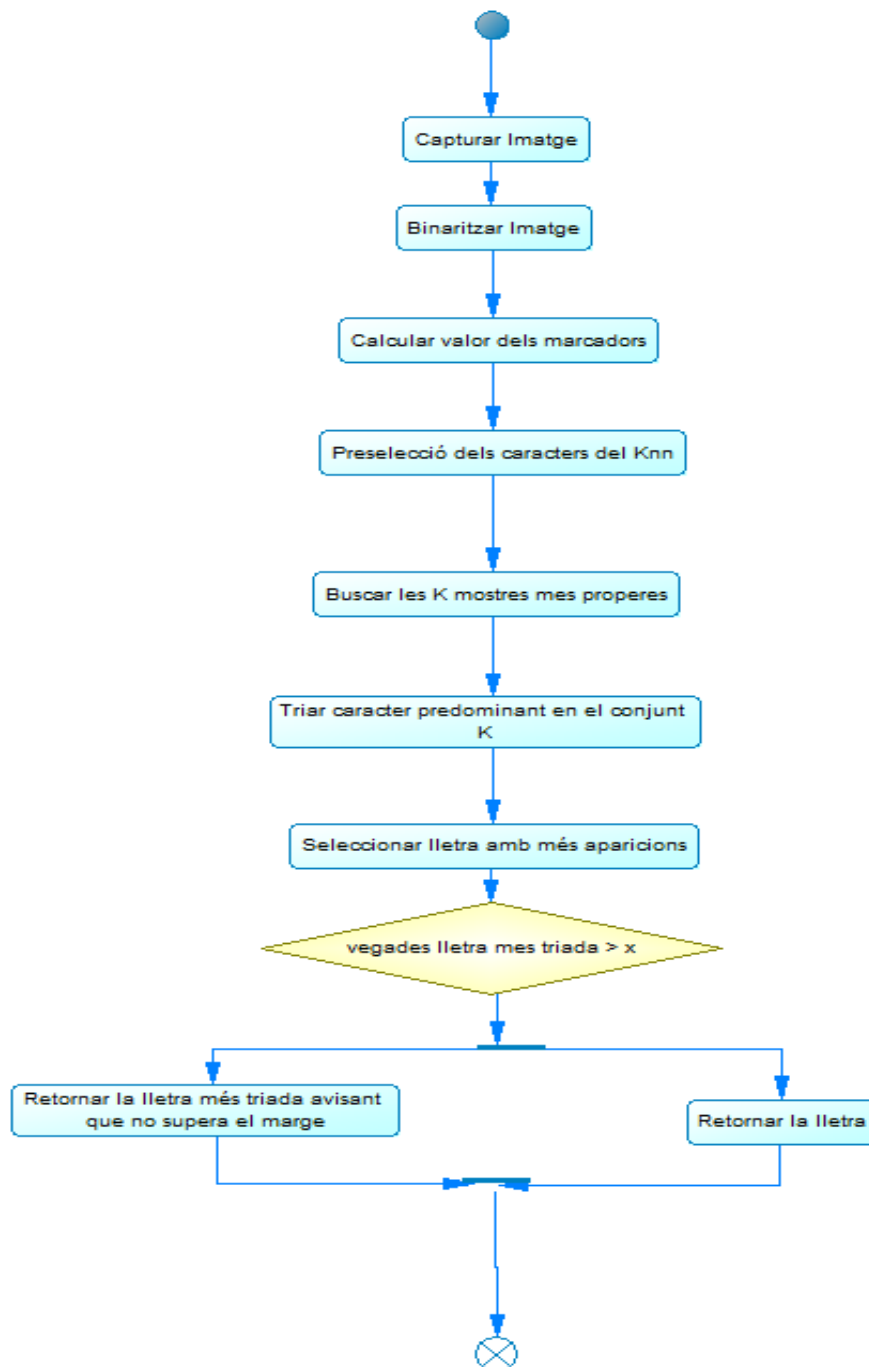


Fig. 4.2 Diagrama d'activitat del processament de signes.

4.5 Diagrama de classes

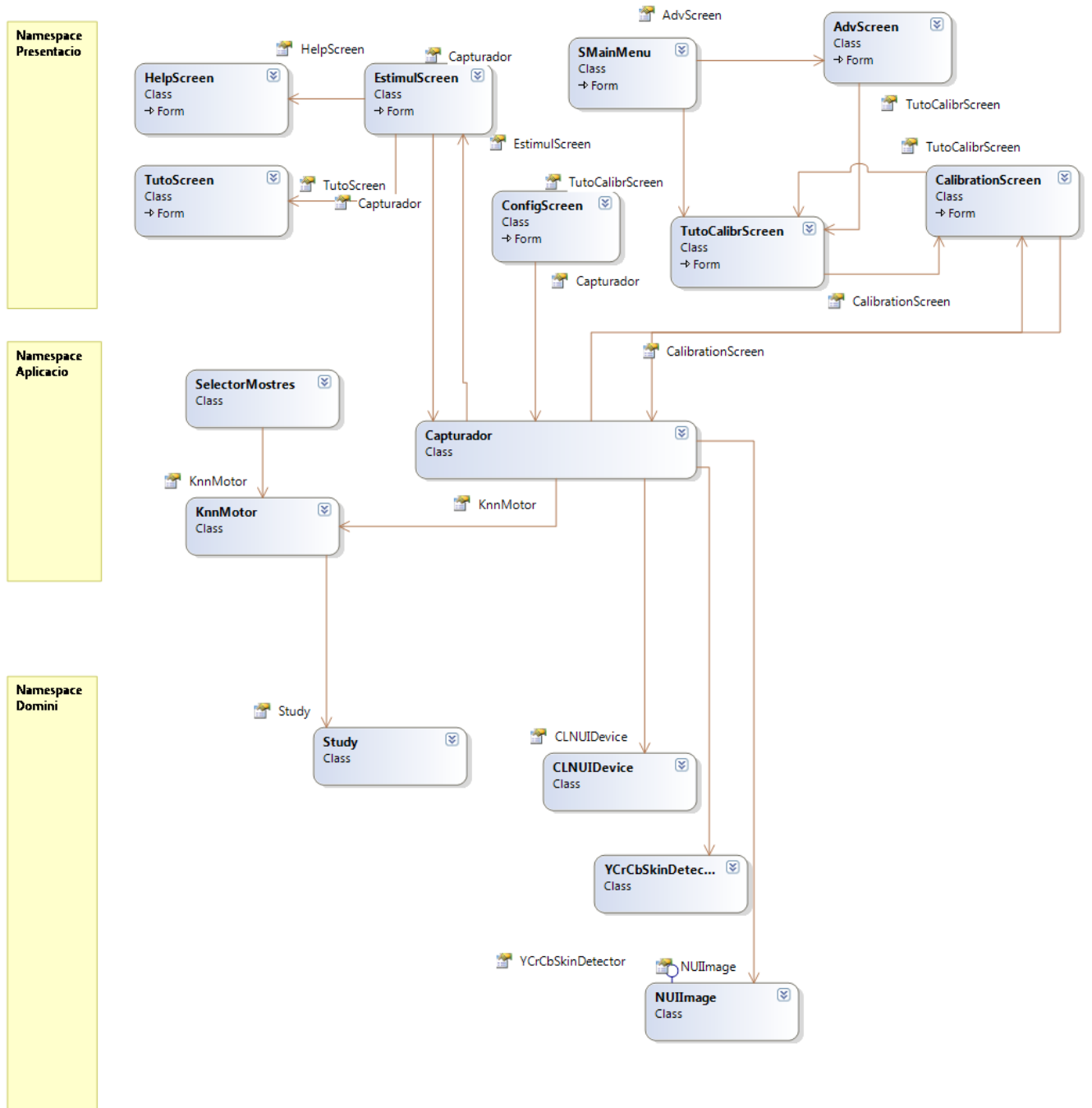


Fig. 4.3 Diagrama de classes

4.6 Metodologia de treball

Per realitzar aquest projecte s'ha realitzat el procés iteratiu i incremental que consisteix en anar afegint funcionalitats noves al software alhora que es realitzen millores sobre la iteració anterior. A continuació es detallen les principals tasques portades a terme en cada iteració

- ✓ Iteració 1 :Creació de la base del reconeixement amb 2 únics marcadors àrea i perímetre
- ✓ Iteració 2 :Revisió dels marcadors anteriors i addició de 2 marcadors més , àrea segmentada i centre gravitatori
- ✓ Iteració 3 :Revisió dels marcadors anteriors eliminant-ne dos i afegint un sistema de preselecció
- ✓ Iteració 4 :Acabar de calibrar el sistema de reconeixement i afegir els dos casos d'us

5. Disseny

5.1 Disseny de la persistència

La persistència es porta a terme usant un sistema de fitxers, s'escull aquest sistema de persistència, ja que es vol que les dades persistides puguin ser modificades amb facilitat pels usuaris del software.

S'utilitzen tres tipus diferents de fitxers:

- Fitxers de format JPG per les imatges, s'emmagatzemen el conjunt d'imatges preses com a mostra per a realitzar el reconeixement, les imatges corresponents als estímuls i les imatges que es prenen durant l'ús del software.

Les imatges corresponents als estímuls són imatges de diferents resolucions ja que aquestes imatges només es mostren durant els estímuls no es realitza cap processament.

Les imatges de les mostres i les capturades durant l'execució de l'aplicació són imatges en la resolució nativa de Kinect 640*480 que s'emmagatzemen binaritzades. Les imatges de les mostres han de ser emmagatzemades amb el nom de la lletra seguit d'un nombre seqüencial de u fins al nombre de mostres, per exemple "A21.jpg".

- Fitxer format "txt" utilitzat per emmagatzemar les paraules a lletrejar ordenades per ordre d'aparició.
- Fitxer format "xml" que conté les dades usades per a la preselecció que són el numero de vèrtexs mínim i màxim, i la orientació per a cadascun dels signes.

```
<Caracter value="A">  
<VertexsMin>4</VertexsMin>  
<VertexsMax>7</VertexsMax>  
<Orientacio>N</Orientacio>  
</Caracter>
```

Fig. 5.1 Tag caràcter del xml de preselecció

5.2 Disseny de la interfície

La interfície s'ha desenvolupat utilitzant c# Forms i el disseny es basa en el de programes de rehabilitació i aprenentatge dissenyats per experts. El programa es pot dividir en les següents pantalles:

Menú Principal

Consta de dos botons un, el superior per iniciar el cas d'us simple i l'inferior pel mode avançat.

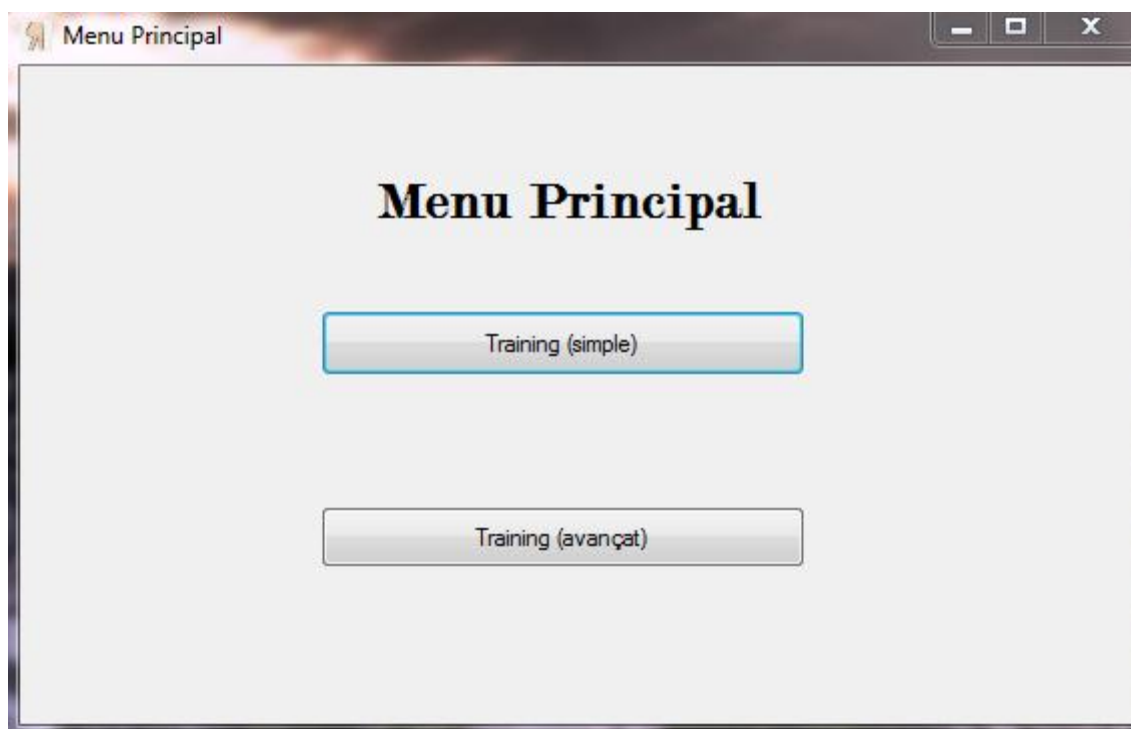


Fig. 5.2 Menú principal

Tutorial calibració

Pantalla on s'explica mitjançant una imatge la postura i la distància adequades per a usar correctament l'aplicació.

També es disposa d'un botó per accedir a les opcions i un altre per iniciar la càrrega del programa.

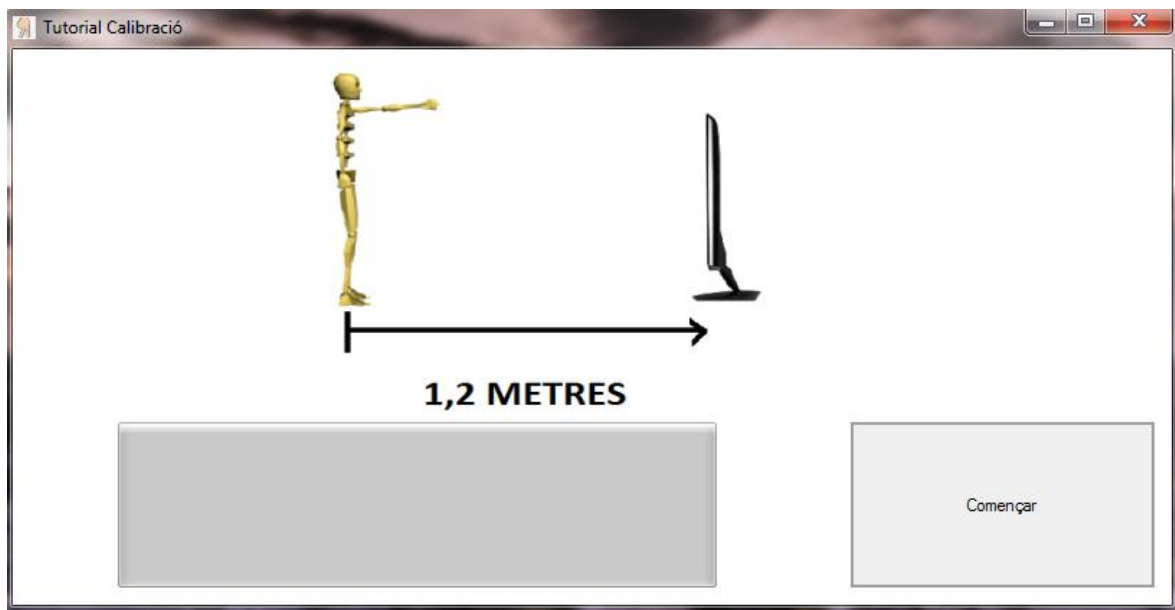


Fig. 5.3 Tutorial calibració

Opcions

Permet escollir la velocitat a la que es realitza l'estímul tenint així més o menys temps per a realitzar-lo.

A més permet modificar la inclinació del dispositiu kinect mitjançant els botons "Amunt" i "Avall"

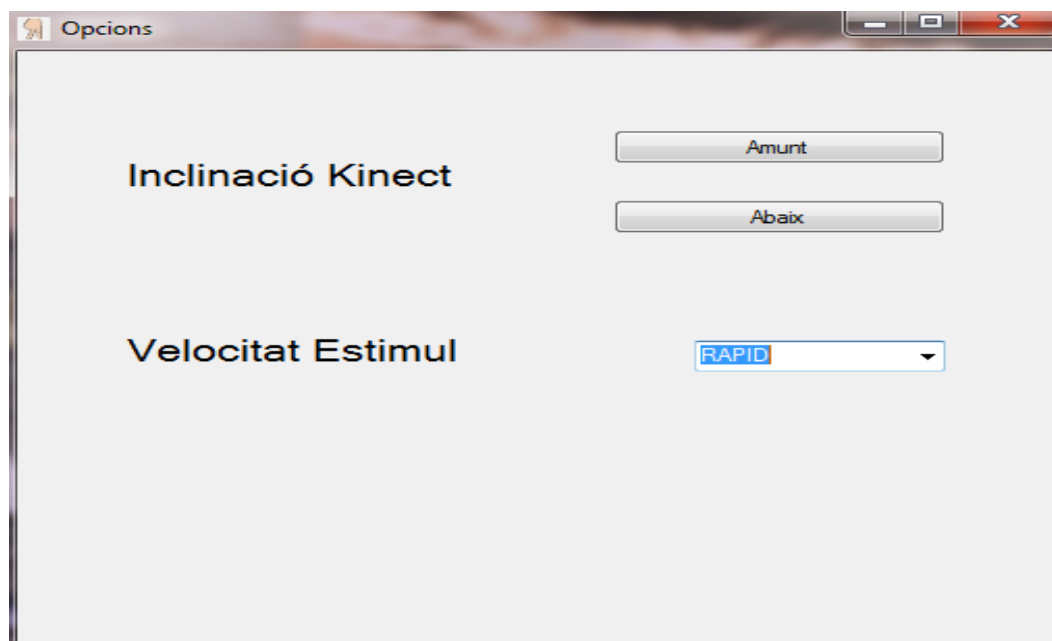


Fig. 5.4 Opcions

Calibració

Pantalla on l'usuari ha de fer coincidir la seva mà amb el contorn verd que es mostra per pantalla; un cop fet això es passa automàticament a l'estímul.

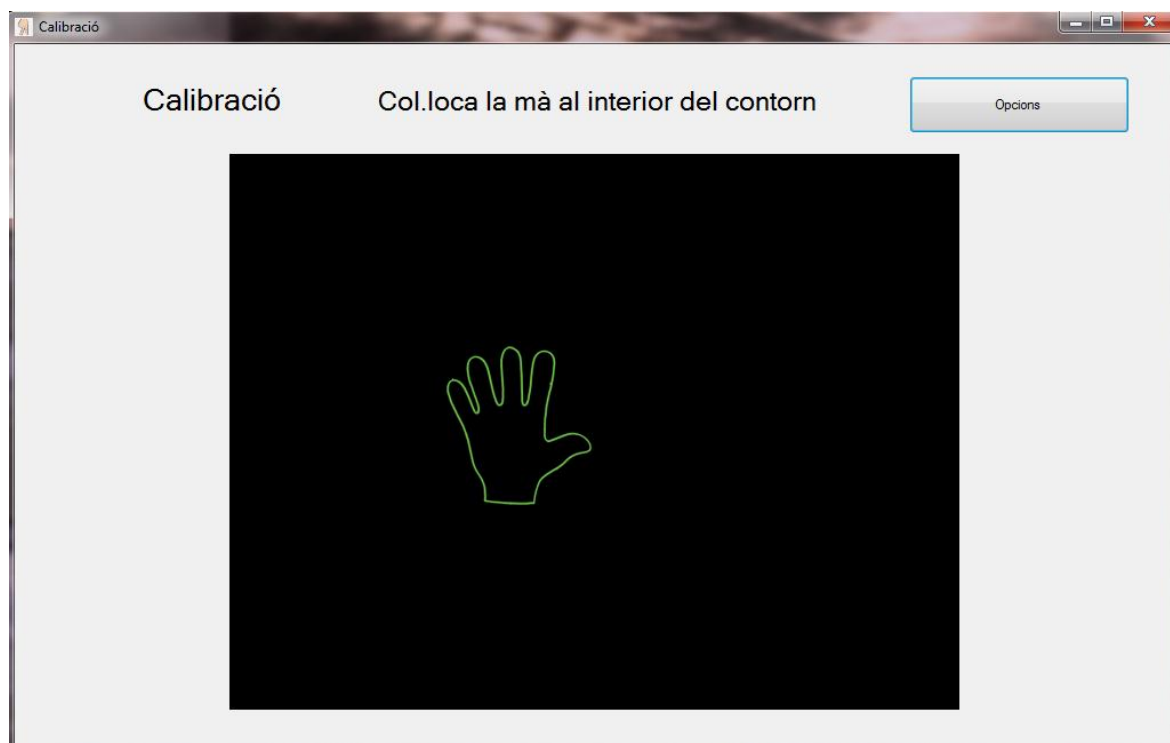


Fig. 5.5 Pantalla Calibració

Estímul

Pantalla on es realitza l'exercici, aquesta pantalla presenta bastanta informació per tal de facilitar la realització de la tasca com :

- La imatge de la paraula a lletrejar
- La paraula a lletrejar escrita per evitar confusions sobre el que mostra la imatge
- La lletra actual escrita
- La lletra actual en llengua de signes per aquelles persones que no coneguin els signes de memòria.
- El botó d'ajuda que explica breument què cal fer
- El botó del tutorial que explica de forma més extensa i detallada en què consisteix la tasca i com s'ha de realitzar.

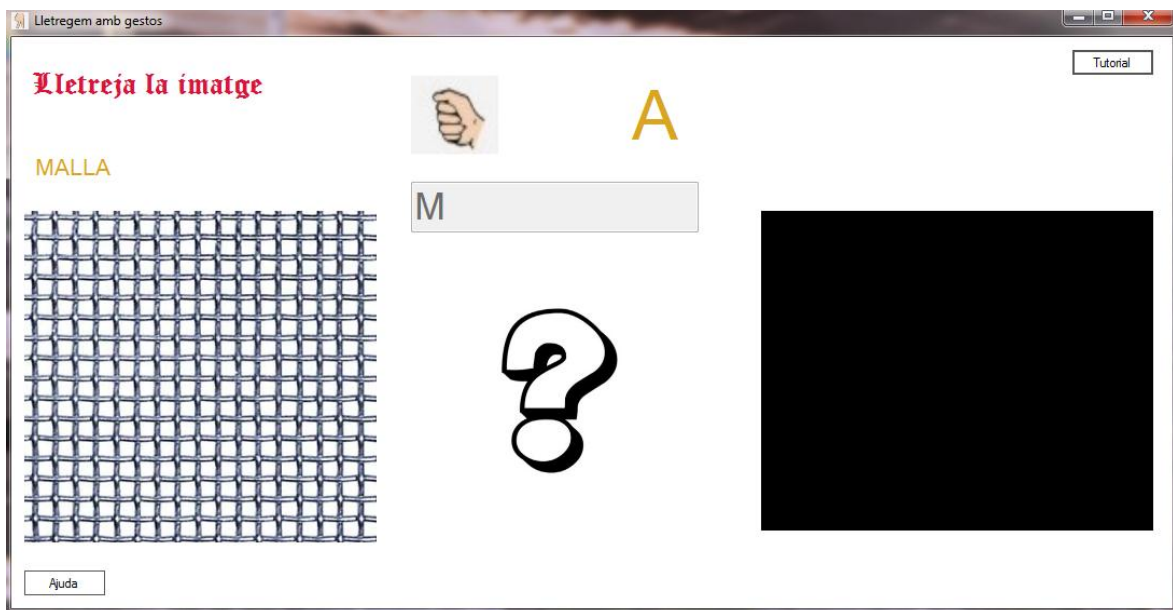


Fig. 5.6 Estímul

Configuració avançada

Pantalla que permet seleccionar una sèrie de paràmetres que en la execució bàsica ja venen predefinits amb el valor que es considera òptim.

Esta enfocada a usuaris interessats en investigació o ampliació del projecte.

Permet definir la K, el numero de mostres per lletra a utilitzar, el marge d'acceptació de l'aplicació i els marcadors.

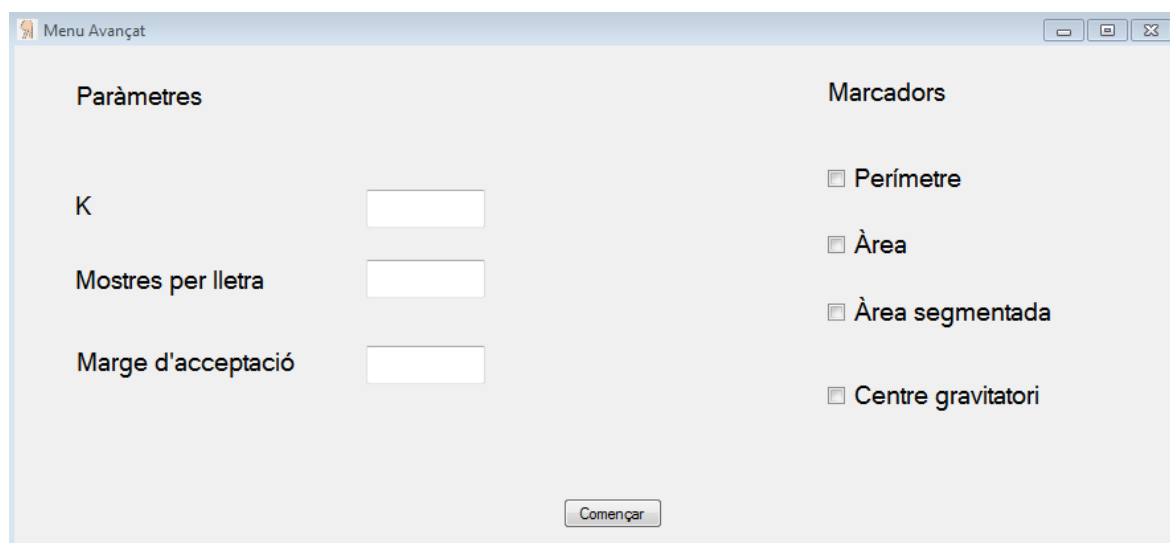


Fig. 5.7 Menú Avançat

6.Desenvolupament

6.1 Configuració de l'entorn de treball

6.1.1 Instal·lació dels drivers

Per al funcionament del projecte és necessari tenir instal·lats els drivers de CLNUI només compatibles amb sistemes Windows, que es poden obtenir de forma gratuïta al web oficial de CodeLaboratories.

És important que el dispositiu Kinect no estigui connectat durant la instal·lació.

El instal·lador no requereix de cap tipus de configuració per part de l'usuari simplement prémer següent fins que finalitzi la instal·lació.

Un cop hagi finalitzat es connecta el dispositiu Kinect i ja esta llest per al seu ús.

6.1.2 Instal·lació de les llibreries

El projecte fa ús de les llibreries externes d'OpenCV que s'usen mitjançant les llibreries d'EmguCV que fan de pont.

Les llibreries d'EmguCV s'inclouen com a referències dintre del Visual Studio i les d'OpenCV s'inclouen dins del directori del projecte.

També cal incloure una referència a les llibreries de CLNUI per tal de poder utilitzar les funcions que aporta el driver de CLNUI.

7. Resultats

7.1 Ratis d'encert

Per tal de comprovar l'efectivitat del reconeixement i la variació que provoquen alguns paràmetres s'han dut a terme una sèrie de proves consistents en observar el tant per cent d'encerts per cada lletra variant una sèrie de paràmetres del reconeixement però mantenint el conjunt de mostres a reconèixer les mostres d'entrenament.

Per determinar quins marcadors són realment útils per al reconeixement s'ha realitzat una prova amb una K fixa de deu, 300 mostres a reconèixer per lletra i 70 mostres d'entrenament per lletra amb totes les combinacions possibles de marcadors.

Amb aquesta prova es va determinar que només el Perímetre, com a marcador més important, i l'àrea segmentada eren realment útils. Aquesta conclusió s'ha extret dels resultats de la prova que determinen que individualment el marcador que aporta millors resultats és el perímetre. L'única manera de millorar els resultats del perímetre individualment és unir-lo amb l'àrea segmentada, aconseguint una millora significativa.

Per determinar el valor òptim de K es realitza una prova amb 300 mostres a reconèixer per lletra i 70 mostres d'entrenament per lletra amb l'àrea segmentada i el perímetre com a marcadors i la K prenent valors entre el 1 i el 30.

Amb aquesta prova es determina que qualsevol K menor de 20 provoca uns ratis d'encert molt similars amb variacions menors al 5%. Per aquest motiu es va determinar que el valor de $K=10$ era el valor òptim ja que en els valors menors de deu en alguns casos si es nota una davallada però en canvi un augment de K per sobre de deu no provoca un augment notori que justifiqui l'augment de temps de processament. En valors per sobre de $K=20$ comença a notar-se una davallada en el rati de reconeixement.

Per últim s'han realitzat proves per a determinar el valor del radi emprat per a descartar les mostres. S'ha observat que aquest valor és molt dependent de les mostres usades com a referència i de les mostres a filtrar a part de ser un valor diferent per a cada lletra degut a que no totes mostren el mateix grau de variabilitat.

7.2 Mètodes de correcció

Els mètodes de correcció que s'han usat han estat principalment dos:

- Eliminació dels marcadors que no contribuïen al reconeixement. Després de la realització del reconeixement sobre un conjunt fixe d'imatges amb totes les combinacions de marcadors. S'ha pogut observar que el marcador de major pes és el perímetre i l'únic marcador que és capaç de contribuir al perímetre positivament es l'àrea segmentada.

Per aquest motiu l'àrea i el centre gravitatori es van descartar en l'execució per defecte quedant només a disposició del usuari avançats que vulguin realitzar proves.

- Addició del sistema de preselecció. Mitjançant un test sobre el Knn utilitzant com a marcadors únicament el perímetre i l'àrea segmentada s'ha determinat que aquest sistema no és capaç de diferenciar dos signes iguals però de diferent orientació. Per aquest motiu s'ha afegit un sistema de preselecció basat en l'orientació del signe.

8. Problemes trobats

Durant la realització del projecte han sorgit diferents problemes, la major part d'ells previstos en el moment de tria del projecte.

Problemes amb el llenguatge de programació.

Aquest tipus de problemes han estat menors ja que precisament es va triar c# per la seva semblança amb els llenguatges coneguts. Tot i ha calgut una petita adaptació tant al entorn de desenvolupament visual estudio com al propi llenguatge.

Problemes amb les llibreries CLNUI.

Aquesta part ha resultat problemàtica degut a la gairebé nul·la documentació d'aquestes llibreries que ha provocat que resultes difícil aconseguir extreure les imatges del Kinect i convertir-les en un format adequat per a C#

Problemes amb les llibreries OpenCV.

Aquests problemes han estat deguts a la desconexença d'aquesta llibreria i a que està enfocada a una gran quantitat de funcions, el que provoca que pugui ser difícil trobar allò que es necessita. Però obviant això la llibreria d'OpenCV ha resultat de gran ajuda en el desenvolupament del projecte.

Problemes amb la captura de la mà

Una mala postura de l'usuari, la profunditat, mida de la mà o l'angle de realització.

La mala postura i la profunditat s'han resolt mitjançant una calibració prèvia, la mida de la mà es resol gràcies a que les mètriques triades gairebé no es veuen afectades per aquest motiu i l'angle de realització està limitat ja que el sistema realitza un reconeixement en dues dimensions.

9. Conclusions

S'ha aconseguit completar els requeriments i objectius de forma satisfactòria, però amb la limitació de no inclusió de la totalitat de les lletres estàtiques que s'ha deixat com una futura ampliació degut a que no s'ha considerat que tingui una aportació essencial al projecte.

D'altra banda es pot lamentar no tenir més temps per tal de continuar investigant en l'àmbit del reconeixement de signes ja que és un àmbit molt extens i complex que requereix d'una gran quantitat de temps per arribar a dominar-lo, a més del temps necessari per aconseguir una mostra gran i variada per a poder definir correctament cada signe.

Però en termes generals tal com marquen els tests s'ha aconseguit un índex d'encert satisfactori.

A nivell personal el treball ha presentat un gran interès degut a basar-se en un àrea com el reconeixement d'imatges que era totalment desconegut a més de treballar amb un altre entorn de treball i llenguatge de programació a part dels coneixements generals que aporta un treball d'aquesta envergadura com la gestió del temps i els recursos, el disseny del software des de zero i la implementació.

10. Futures ampliacions

Detecció de lletres en moviment:

Realitzar la detecció estàtica de les lletres que no siguin iguals a una altre però en moviment com és el cas de la W.

Per les que siguin iguals a un altre signe però en moviment com la U i la V es podria fer un cas específic en el que si es detectes que és per exemple una U comprovar si es en moviment o no mitjançant per exemple la variació de la orientació o determinar-ho a la preselecció

Substitució del Kinect per una Web Cam: per fer això només caldria modificar la part del captador que recull la imatge del Kinect per que reculli la imatge de la web cam y el rang de colors a seleccionar que s'haurien de correspondre amb els colors de la pell.

Un cop fet això el sistema de reconeixement funcionaria correctament però amb algunes restriccions. Caldria utilitzar la captura sobre un fons de color uniforme i clarament diferenciat del color de la pell. L'usuari ha de portar roba fosca i ha de fer els signes des de fora del pla de captura.

Compatibilitat amb altres llengües de signes: per aconseguir adaptar-lo a una altra llengua de signes caldria modificar les mostres corresponents a cada lletra i recalculer els valors de preselecció.

Això permetria fer-lo compatible per exemple amb la ASL o la Llengua de signes castellana.

Desenvolupament d'altres estímuls: Desenvolupar activitats que permetin com per exemple el joc del penjat , o un joc de paraules creuades.

11. Referències:

- [1] philipjfry.cephis.uab.cat/signem/index.php , UAB, Guia bàsica per a la comunicació en llengua de signes catalana. 30 febrer 2012.
- [2] blocs.xtec.cat/cinemaencurstrespins0708/2008/02/18/alfabet-dactilologic/, Imatge alfabet dactilològic. 18 febrer 2012.
- [3] emgu.com/wiki/index.php/Main_Page, EmguCV, Documentació EmguCv. 20 Febrer 2012.
- [4] opencv.willowgarage.com/wiki/, OpenCv, Documentació OpenCV. 20 Febrer 2012.
- [5] xbox.com/es-ES/kinect , Microsoft, Característiques del Kinect. 2012.
- [6] event.asus.com/wavi/product/WAVI_Xtion.aspx, Asus, Característiques del Wavi Xtion. 20 Febrer 2012.
- [7] microsoft.com/en-us/kinectforwindows/develop/overview.aspx, Microsoft, SDK Kinect per Windows. 20 Febrer 2012.
- [8] codelaboratories.com/nui, CLNUI Platform, Documentació llibreria CLNUI, 10 Febrer 2012.

Escola Universitària Politécnica de Mataró

Centre adscrit a:



UNIVERSITAT POLITÈCNICA
DE CATALUNYA

Enginyeria Tècnica d'Informàtica de Gestió

Aprenentatge de la llengua de signes catalana amb Kinect

Estudi econòmic

Marc Górriz
PONENT: Alfons Palacios

PRIMAVERA 2012



TecnoCampus
Mataró-Maresme

Índex.

1. Cost del prototip.	1
1.1. Costos de recursos humans.	1
1.2. Amortització equips, instrumental i software.	1
1.3. Cost de fabricació del prototip.	2

1. Cost del prototip.

1.1. Costos de recursos humans.

<u>Concepte</u>	<u>Hores</u>	<u>Preu/hora (€)</u>	<u>Total (€)</u>
Estudi i documentació (Enginyer sènior)	60	30	1.800
Testeig i millora (Enginyer sènior)	80	30	2.400
Desenvolupament(Enginyer júnior)	200	20	4.000
Redacció memòria (Administratiu)	50	15	750
TOTAL RECURSOS HUMANS			8.950

1.2. Amortització equips, instrumental i software.

<u>Equip utilitzat</u>	<u>Vida util</u> <u>(hores)</u>	<u>Preu(€)</u>	<u>Hores</u> <u>d'utilització</u>	<u>Preu/hora(€)</u>	<u>Total(€)</u>
<u>Equips i programari informàtic</u>					
Ordinador	6000	700	390	0,12	46,8
Visual Studio 2010	8000	500	200	0,06	12
Power Designer	3000	6000	10	2,00	20,0
Microsoft Office	8000	379	50	0,05	2,5
Kinect	4000	99	280	0,02	5,6
TOTAL AMORTITZACIONS					86,9

1.3. Cost de fabricació del prototip.

Costos de recursos humans	8.950€
Costos d'amortització	86,9 €
Subtotal	9.036,9 €
Despeses indirectes (19%)	1.717,01€
TOTAL	10.753,91€

Escola Universitària Politécnica de Mataró

Centre adscrit a:



UNIVERSITAT POLITÈCNICA
DE CATALUNYA

Enginyeria Tècnica d'Informàtica de Gestió

Aprenentatge de la llengua de signes catalana amb Kinect

Annexos

Marc Górriz
PONENT: Alfons Palacios

PRIMAVERA 2012



TecnoCampus
Mataró-Maresme

Índex.

Annex I. Codi captura de la mà.	1
Annex II. Codi Knn.	3
Annex III. Contingut del DVD-ROM.	5

Annex I. Codi captura de la mà.

```
public Capturador(EstimulScreen entrada, CalibrationScreen
    entradaC, Dictionary<String, int> parametres)
{
    calibrat = false;
    marcadors = new bool[4]{false,false,false,false};
    motorKnn = new KnnMotor();
    if (parametres != null)
    {
        int num = -1;
        parametres.TryGetValue("k", out num);
        if (num != -1) { kValue = num; }
        else
        {
            kValue = 10;
        }

        parametres.TryGetValue("numMostres", out num);
        if (num != -1) { motorKnn.setNumMostres(num); }
        else
        {
            motorKnn.setNumMostres(70);
        }

        parametres.TryGetValue("acceptacio", out num);
        if (num != -1) { margeAcceptacio = num; }
        else
        {
            margeAcceptacio = 15;
        }
        parametres.TryGetValue("area", out num);
        if (num == 1) { marcadors[0] = true; }

        parametres.TryGetValue("perimetre", out num);
        if (num == 1) { marcadors[1] = true; }
        parametres.TryGetValue("centreG", out num);
        if (num == 1) { marcadors[2] = true; }
        parametres.TryGetValue("areaSeg", out num);
        if (num == 1) { marcadors[3] = true; }

    }
    else { kValue = 10;
        motorKnn.setNumMostres(70);
        margeAcceptacio = 15;
        marcadors[1] = true;
        marcadors[3] = true;
    }
}
```

```
margeMostres = 70;
this.entrada = entrada;
this.entradaC = entradaC;
motor = CLNUIDevice.CreateMotor(CLNUIDevice.GetDeviceSerial(0));
camera = CLNUIDevice.CreateCamera(CLNUIDevice.GetDeviceSerial(0));
CLNUIDevice.StartCamera(camera);
colorImage = new NUIImage(640, 480);
YCrCb_min = new Ycc(40, 50, 150);
YCrCb_max = new Ycc(255, 150, 255);

lletraPerCaptura = new List<string>();
ProgressBar pB = entradaC.getTutoCalibrScreen().getProgressBar1();
pB.Step = pB.Maximum / numCaracters+1;
motorKnn.cargarTodoVarianca(pB);
caractersCount = new int[numCaracters];
}
```

Annex II. Codi Knn.

```

public List<String> KnnComplertVarianca(Bitmap entrada, int k, bool area,
    bool perimetre, bool centreGravitatori, bool areaSegmentada)
{
    //Comprobar que les imatges de traininc estan carregades
    if (!carregat)
    { this.cargarTodoVarianca(); }

    List<Study> llista;
    List<String> resultats = new List<string>();

    int lletra;
    Study estudiEntrada, estudio;
    //Taula que conte els K estudis més propers
    Study[] tags = new Study[k];

    for (int i = 0; i < k; i++)
    {
        tags[i] = new Study("", 0, 0, 1000000000000000000, new List<double>(), new Point());
    }
    Image<Gray, Byte> iMostra = new Image<Gray, byte>(entrada);
    Study stMostra = this.estudiDeMostra(iMostra, "");
    estudiEntrada = stMostra;
    //Si la mostra es correcte es comencen a calcular les distancies
    if (!stMostra.lletra.Equals("error"))
    {
        //Obtenir els caracters contra els que es farà el Knn
        caracters=this.getCaractersFiltrats(this.biggestContourEntrada);
        caractersCount=new int[caracters.Length];
        for (int i = 0; i < caractersCount.Length; i++) { caractersCount[i] = 0; }

        //Emplenar la taula tags amb els K mes propers
        for (int j = 0; j < caracters.Length; j++)
        {
            trainData.TryGetValue(caracters[j], out llista);
            for (int i = 0; i < numMostres; i++)
            {
                estudio = llista.ElementAt(i);
                estudio.dist = this.calcularDistEuclComplertVarianca(estudio, estudiEntrada, area,
                    perimetre, centreGravitatori, areaSegmentada, variances);

                if (estudio.dist < tags[k - 1].dist)
                {
                    tags[k - 1] = estudio;
                    this.Burbuja(tags);
                }
            }
        }
    }
}

```

```
//contar la quantitat de cada que hi han al grup tags
for (int p = 0; p < tags.Length; p++)
{
    lletra = this.posCaracter(tags[p].lletra);
    if(lletra>=0)
        this.charactersCount[lletra]++;
}

for (int i = 0; i < tags.Length; i++)
{
    resultats.Add(tags[i].lletra);
}

resultats.Add(lletraMax(charactersCount));
return resultats;
}
else { resultats.Add("error"); return resultats; }
}
```

Annex III. Contingut del DVD-ROM.

- Documentació del projecte (memòria, estudi econòmic i annexos).
- Llibreries utilitzades en el projecte (EmguCV, OpenCV) .
- Drivers necessaris (CLNUIPlatform).
- Software desenvolupat, solució C# per Visual Studio 2010.
- Software desenvolupat amb instalador Windows.